The following outline shows the sequence of computations used in the South Florida Water Management Model. Related subroutines and functions are given below each major model task. A brief description of all subroutines is also given at the end of this section. Some key points are needed in interpreting the SFWMM's program structure are:

1. The model runs on a fixed time step of one day and a fixed grid size of 2 miles by 2 miles. The simulation period ranges from a minimum of one entire month to a maximum of 31 years. Local phenomena, such as delineation of floodplains or groundwater cones of depression, are beyond the degree of resolution provided by the model. The overland flow algorithm uses a smaller time-step in order to maintain model stability. Refer to Sec. 2.3 for more details.

2. Hydrologic processes are simulated independently within a single time step. The need to simulate feedback between related processes during the day, e.g. surface and subsurface processes, is minimized by arranging the sequence of calculations from more transient processes such as infiltration, to fairly static processes such as groundwater flow. This scheme is beneficial in two ways: (1) model execution time is reduced compared to algorithms that involve more complex (iterative) schemes; and (2) debugging can be isolated to specific subroutines or functions.

3. The management routines are interconnected with the hydrologic components as necessary within the two time loops (monthly and daily) specified below.

4. The model is CPU- as well as IO- intensive. The UNIX workstation environment provides the best environment, to date, because it has the number-crunching power as well as enormous disk space required to complete a model run. The model is not meant to be a "point-and-click" application. The model is trying to simulate a very complex water resources system. For this reason, an in-depth knowledge of the natural and management processes associated with the system is necessary before using the model.

5. The reader may want to refer to the flowchart shown in Sec. 1.3 (Fig. 1.3.3) for guidance.

GENERAL PROGRAM ORGANIZATION

main program

- **A.** open i/o files
- **B.** read time-invariant input data
  - indata, cnldata, trigger_input, asrinput
- **C.** execute monthly time loop

monthly time loop:

1. reduce public water supply pumpage by a fraction determined at the end of a tracking period (usually at the end of a month)
- reduce_wellq

2. execute daily time loop
3. identify which zones require water restriction and set appropriate cutback level to be implemented in the following month
   - set_cutback

daily time loop:

a. calculate daily net irrigation supply and corresponding shortage
- lec_set_cutback

b. compute levee seepage
- lvseep

c. compute demand / runoff in the EAA
- agarea

d. meet EAA demand, if any, from local storage facilities
- ws_from_res, asr

e. do calculations for LOSA cutback scheme
- ssm

f. calculate volume of water required to maintain canals in the LEC
- wsneeds

g. determine structure flows which are explicit functions of cell and/or canal stages based on operating rules related to water supply, flood control and environmental policies
- route, locwslwdd, lakewca, wcaout

h. update appropriate nodal stage based on **g.**
- knflows

i. establish canal equilibrium stage by considering canal budget components
- chnlf

j. overland flow
- ovlnf

k. infiltration / percolation
- ovlnf, gwf

l. evapotranspiration
- etcomp, lec_et_comp

m. groundwater flow
- gwf

n. compute outflow from reservoirs, such as STAs, primarily for flood control purposes
- stastor, staout, staws

o. keep track of canal and/or nodal stages at designated trigger locations at appropriate days of the month
- sum_trig_heads